Logistic Regression:
Assumptions, Limitations And Applications

Ahmed BADI
ahmedbadi905@gmail.com
linkedin.com/in/badi-ahmed

December 5, 2025

Abstract

Logistic regression is one of the most widely used statistical and machine learning techniques for modeling binary outcomes. This article provides a clear and comprehensive overview of the logistic regression framework, including its mathematical formulation, underlying assumptions, estimation through maximum likelihood, and interpretation of model coefficients. Practical aspects such as model evaluation, threshold selection, and common pitfalls (e.g., multicollinearity, class imbalance) are also discussed. Furthermore, the article highlights extensions of logistic regression, including multinomial and regularized variants, and illustrates its applications across domains such as healthcare, finance, and marketing. The objective is to offer both theoretical insight and practical guidance for students, researchers, and practitioners working with classification problems.

Keywords: Logistic Regression, Binary Classification, Maximum Likelihood Estimation, Odds Ratio, Model Evaluation, Multicollinearity, Regularization, Machine Learning, Predictive Modeling.

1 Introduction

In today's data-driven world, we often face problems where we need to predict whether something will happen or not. Will a customer buy a product? Will a patient develop a disease? Will an email be spam? These yes-or-no questions are everywhere in business, medicine, science, and everyday life.

Logistic regression is a powerful tool designed specifically for these kinds of problems [1]. Despite its name containing the word "regression," it's actually a classification method. The technique has stood the test of time since its introduction in the 1940s [2] and remains one of the most popular approaches in statistics and machine learning today.

What makes logistic regression so special? First, it's interpretable. Unlike many complex machine learning algorithms that work like black boxes, logistic regression tells us exactly how each factor influences our prediction. Second, it's efficient and works well even with smaller datasets. Third, it provides not just predictions but also probabilities, which helps us understand how confident we should be about each prediction [3].

In this article, we'll explore logistic regression from the ground up. We'll start with the intuition behind the method, dive into the mathematics that make it work, learn how to build and evaluate models, and discover practical tips for using it effectively. Whether you're a student learning statistics, a data scientist building predictive models, or a researcher analyzing data, this guide will give you both the theoretical understanding and practical knowledge you need.

2 The Problem: From Linear Regression to Classification

2.1 Why Linear Regression Doesn't Work for Classification

Let's start by understanding why we need a special method for classification. Imagine you're trying to predict whether students will pass an exam based on their study hours. With linear regression, you might write:

$$y = \beta_0 + \beta_1 x + \epsilon \tag{1}$$

where y is the outcome (pass/fail) and x is study hours.

The problem? Linear regression can predict values outside the range [0,1]. You might get predictions like 1.5 or -0.3, which make no sense for probabilities. Additionally, linear regression assumes the relationship between predictors and outcome is a straight line, but in reality, the probability of passing changes differently at different study hour levels [4].

2.2 Enter the Logistic Function

The solution is to use a function that naturally stays between 0 and 1. This is where the logistic function (also called the sigmoid function) comes in:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

This can also be written in an alternative form:

$$\sigma(z) = \frac{e^z}{1 + e^z} \tag{3}$$

This beautiful S-shaped curve has exactly the properties we need. When z is very negative, $\sigma(z)$ approaches 0. When z is very positive, $\sigma(z)$ approaches 1. And it smoothly transitions between these extremes.

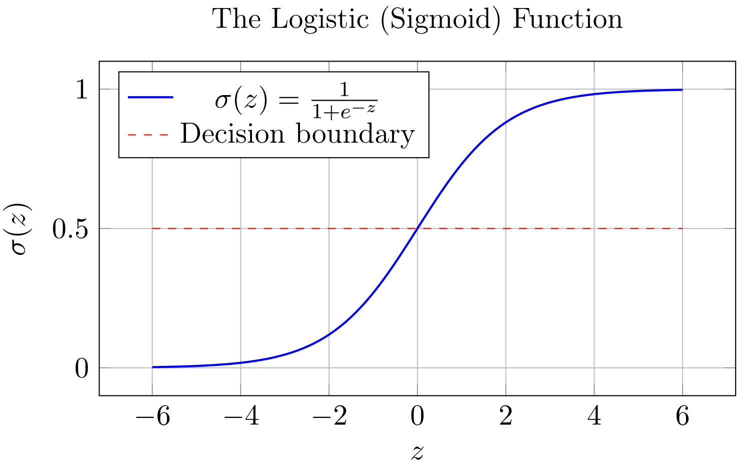

Figure 1: The sigmoid function transforms any real number into a probability between 0 and 1

3 Types of Logistic Regression

Before diving deeper into the mathematics, it's important to understand that logistic regression comes in different flavors depending on the nature of your outcome variable [5].

3.1 Binomial (Binary) Logistic Regression

This is the most common type and what most people mean when they say "logistic regression." It's used when the outcome has exactly two possible categories. Examples include:

- Will a customer buy or not buy?
- Is an email spam or not spam?
- Does a patient have the disease or not?
- Pass or fail?

The model predicts the probability of belonging to one class (typically labeled as 1) versus the other (labeled as 0).

3.2 Multinomial Logistic Regression

When your outcome has three or more categories that have no inherent order, you need multinomial logistic regression. Examples include:

- Classifying animals into categories (cat, dog, bird)
- Predicting which product a customer will choose (Product A, B, or C)
- Identifying types of diseases (Disease Type 1, 2, or 3)

Instead of the sigmoid function, multinomial logistic regression uses the softmax function, which generalizes the sigmoid to multiple classes. For K classes, the softmax function is:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{4}$$

where K represents the number of elements in the vector z and i, j iterate over all the elements in the vector.

The probability that an observation belongs to class c is then:

$$P(Y = c|\mathbf{X} = \mathbf{x}) = \frac{e^{\mathbf{w}_c^T \mathbf{x} + b_c}}{\sum_{k=1}^{K} e^{\mathbf{w}_k^T \mathbf{x} + b_k}} \tag{5}$$

where each class has its own set of weights $\mathbf{w}_c$ and bias b_c.

3.3 Ordinal Logistic Regression

This type is used when your outcome has three or more categories with a natural ordering or ranking. Examples include:

- Customer satisfaction ratings (very dissatisfied, dissatisfied, neutral, satisfied, very satisfied)
- Disease severity (mild, moderate, severe)
- Education level (high school, bachelor's, master's, PhD)
- Product ratings (1 star, 2 stars, 3 stars, 4 stars, 5 stars)

The key difference from multinomial regression is that ordinal regression respects the ordering of categories. If we're predicting satisfaction levels, the model understands that "satisfied" is closer to "very satisfied" than to "very dissatisfied."

Ordinal logistic regression uses cumulative probabilities and assumes that the effect of predictors is the same across all category thresholds (the proportional odds assumption).

4 Key Terminology in Logistic Regression

Before we proceed further, let's clarify some important terms that are frequently used when discussing logistic regression:

- **Independent Variables (Features/Predictors)**: These are the input variables or features used to make predictions. For example, in predicting disease, independent variables might include age, weight, blood pressure, and cholesterol level.
- **Dependent Variable (Target/Response)**: This is the outcome variable we aim to predict. In logistic regression, it must be categorical. For binary logistic regression, it takes values like 0/1, Yes/No, or True/False.
- **Logistic Function (Sigmoid)**: The mathematical function that transforms any real-valued input into a probability between 0 and 1, creating the characteristic S-shaped curve.
- **Odds**: The ratio of the probability of an event occurring to the probability of it not occurring. If $p = 0.8$, then odds $= 0.8/0.2 = 4$, meaning success is 4 times as likely as failure. Note that odds differ from probability.
- **Log-Odds (Logit)**: The natural logarithm of the odds. In logistic regression, the log-odds are modeled as a linear combination of the predictors. This is what makes the relationship "linear" in logistic regression.
- **Coefficients (Weights)**: The parameters (β values) estimated by the model that quantify how strongly each independent variable affects the log-odds of the outcome. A positive coefficient increases the probability of the positive class, while a negative coefficient decreases it.
- **Intercept (Bias)**: The constant term β_0 in the model, representing the log-odds when all independent variables equal zero. It shifts the decision boundary.
- **Maximum Likelihood Estimation (MLE)**: The statistical method used to estimate the coefficients by finding the parameter values that maximize the probability (likelihood) of observing the actual data given the model.
- **Decision Threshold**: The probability cutoff (typically 0.5) used to convert predicted probabilities into binary class predictions. Values above the threshold are classified as positive (1), and values below as negative (0).

5 Mathematical Foundation

5.1 The Logistic Regression Model

In logistic regression, we model the probability that an observation belongs to the positive class (typically denoted as $y = 1$). Given a vector of input features $\mathbf{x} = (x_1, x_2, \ldots, x_p)$ and parameters $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)$, the model is [6]:

$$P(Y = 1|\mathbf{x}) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)}} \tag{6}$$

We can write this more compactly. Let $z = \beta_0 + \boldsymbol{\beta}^T \mathbf{x}$ be the linear combination of inputs. Then:

$$P(Y = 1|\mathbf{x}) = \sigma(z) = \frac{1}{1 + e^{-z}} \tag{7}$$

5.2 The Logit Transform: Understanding the Odds

To better understand what logistic regression is doing, let's introduce the concept of odds. If the probability of an event is p, the odds are:

$$\text{Odds} = \frac{p}{1-p} \tag{8}$$

For example, if the probability of passing is 0.75, the odds are $\frac{0.75}{0.25} = 3$, meaning success is 3 times as likely as failure.

The logit function is the natural logarithm of the odds:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) \tag{9}$$

Here's the key insight: in logistic regression, the logit is a linear function of the predictors:

$$\log\left(\frac{P(Y=1|\mathbf{x})}{P(Y=0|\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p \tag{10}$$

This means that while the probability has a nonlinear relationship with the predictors, the log-odds has a perfectly linear relationship. This is why we call it logistic regression!

5.3 Interpreting Coefficients: Odds Ratios

One of the most powerful features of logistic regression is that coefficients have a clear interpretation through odds ratios. When we increase predictor x_j by one unit (holding all other predictors constant), the odds multiply by e^{β_j} [1].

Mathematically:

$$\text{Odds Ratio} = e^{\beta_j} \tag{11}$$

For example:

- If $\beta_1 = 0.5$, then $e^{0.5} \approx 1.65$, meaning each unit increase in x_1 multiplies the odds by 1.65 (a 65% increase)
- If $\beta_2 = -0.3$, then $e^{-0.3} \approx 0.74$, meaning each unit increase in x_2 multiplies the odds by 0.74 (a 26% decrease)
- If $\beta_3 = 0$, then $e^0 = 1$, meaning x_3 has no effect on the odds

6 Parameter Estimation: Maximum Likelihood

6.1 The Likelihood Function

Unlike linear regression where we can find parameters using simple formulas, logistic regression requires an iterative optimization method. We use maximum likelihood estimation (MLE) [7].

For a dataset with n observations (x_i, y_i) where $y_i \in \{0, 1\}$, the likelihood of observing the data given parameters $\boldsymbol{\beta}$ is:

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} P(Y = y_i|\mathbf{x}_i) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i} \tag{12}$$

where $p_i = P(Y = 1|\mathbf{x}_i)$.

Taking the logarithm (which doesn't change the location of the maximum), we get the log-likelihood:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} [y_i \log(p_i) + (1-y_i)\log(1-p_i)] \tag{13}$$

We can rewrite this as:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} [y_i \log(p_i) + \log(1-p_i) - y_i \log(1-p_i)] \tag{14}$$

$$= \sum_{i=1}^{n} \left[\log(1-p_i) + y_i \log\left(\frac{p_i}{1-p_i}\right)\right] \tag{15}$$

$$= \sum_{i=1}^{n} \left[-\log(1+e^{\mathbf{w}^T\mathbf{x}_i+b}) + y_i(\mathbf{w}^T\mathbf{x}_i + b)\right] \tag{16}$$

6.2 Optimization: Finding the Best Parameters

To find the parameters that maximize the log-likelihood, we typically use numerical optimization algorithms like Newton-Raphson or gradient descent [8]. The gradient of the log-likelihood with respect to parameter β_j is:

$$\frac{\partial \ell}{\partial \beta_j} = -\sum_{i=1}^{n} \frac{1}{1+e^{\mathbf{w}^T\mathbf{x}_i+b}} \cdot e^{\mathbf{w}^T\mathbf{x}_i+b} \cdot x_{ij} + \sum_{i=1}^{n} y_i x_{ij} \tag{17}$$

$$= -\sum_{i=1}^{n} p_i x_{ij} + \sum_{i=1}^{n} y_i x_{ij} \tag{18}$$

$$= \sum_{i=1}^{n} (y_i - p_i) x_{ij} \tag{19}$$

The algorithm iteratively updates the parameters using gradient ascent:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + \alpha \frac{\partial \ell}{\partial \boldsymbol{\beta}} \tag{20}$$

where α is the learning rate.

7 Model Evaluation and Diagnostics

7.1 Making Predictions

Once we've estimated the parameters, we can make predictions. For a new observation $\mathbf{x}_{new}$, we calculate:

$$\hat{p} = P(Y=1|\mathbf{x}_{new}) = \frac{1}{1+e^{-(\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^T \mathbf{x}_{new})}} \tag{21}$$

To convert this probability to a binary prediction, we use a threshold (typically 0.5):

$$\hat{y} = \begin{cases} 1 & \text{if } \hat{p} \geq 0.5 \\ 0 & \text{if } \hat{p} < 0.5 \end{cases} \tag{22}$$

7.2 The Confusion Matrix

A confusion matrix summarizes the performance of our classifier:

		Predicted	
		Positive	Negative
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Table 1: Confusion matrix structure

From this, we derive several useful metrics [9]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{23}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{24}$$

$$\text{Recall (Sensitivity)} = \frac{TP}{TP + FN} \tag{25}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{26}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{27}$$

7.3 ROC Curve and AUC-ROC

The Receiver Operating Characteristic (ROC) curve plots the true positive rate (sensitivity) against the false positive rate (1 - specificity) at various threshold settings. The area under this curve (AUC-ROC) provides a single number summarizing model performance across all possible thresholds [9].

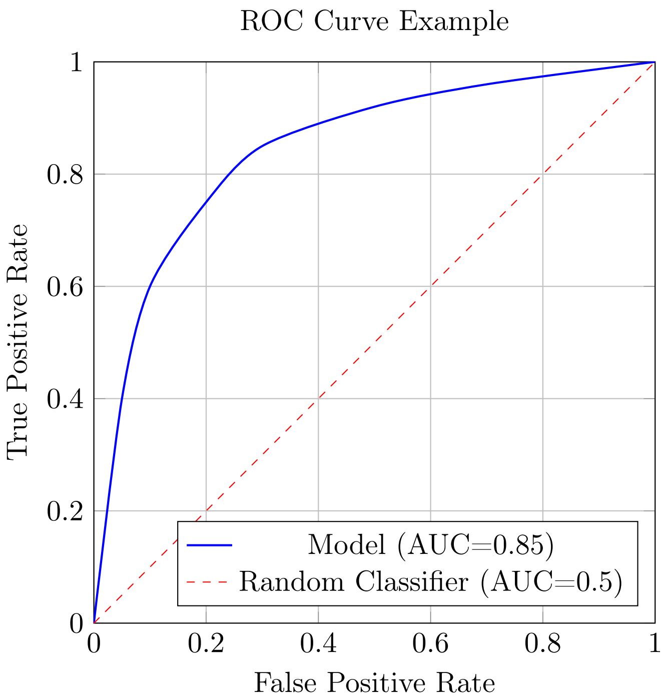

Figure 2: ROC curve showing model performance. The closer to the top-left corner, the better.

An AUC of 0.5 means the model is no better than random guessing, while an AUC of 1.0 represents perfect classification. Generally:

- AUC = 0.9-1.0: Excellent
- AUC = 0.8-0.9: Good
- AUC = 0.7-0.8: Fair
- AUC = 0.6-0.7: Poor
- AUC = 0.5-0.6: Fail

7.4 Precision-Recall Curve and AUC-PR

While ROC curves are widely used, they can be misleading when dealing with highly imbalanced datasets. In such cases, the Precision-Recall (PR) curve provides a more informative picture of model performance.

The PR curve plots precision against recall at various thresholds. The area under the precision-recall curve (AUC-PR) summarizes the trade-off between precision and recall across different threshold values.

AUC-PR is particularly useful when:

- The positive class is rare (class imbalance)
- False positives are particularly costly
- You care more about the positive predictions being correct

For example, in fraud detection where fraud cases are rare (say 1% of transactions), a model that predicts "no fraud" for everything would have 99% accuracy but would be useless. The PR curve would reveal this problem more clearly than the ROC curve [10].

8 Linear Regression vs Logistic Regression

Since logistic regression has "regression" in its name, newcomers often confuse it with linear regression. Let's clarify the key differences [3]:

Linear Regression	**Logistic Regression**
Used to predict continuous dependent variables	Used to predict categorical dependent variables
Solves regression problems	Solves classification problems
Predicts values of continuous variables (e.g., house price, temperature)	Predicts values of categorical variables (e.g., 0 or 1, Yes or No)
Finds the best fit straight line	Finds an S-shaped curve (sigmoid)
Uses Ordinary Least Squares (OLS) for parameter estimation	Uses Maximum Likelihood Estimation (MLE) for parameter estimation
Output can be any real number ($-\infty$ to $+\infty$)	Output is bounded between 0 and 1 (probability)
Assumes a linear relationship between dependent and independent variables	Models a linear relationship between independent variables and log-odds (not the probability directly)
Some multicollinearity between independent variables is tolerable	Requires little to no multicollinearity between independent variables
Evaluated using metrics like R-squared, MSE, RMSE	Evaluated using accuracy, precision, recall, F1-score, AUC-ROC
Example: Predicting salary based on years of experience	Example: Predicting whether a student will pass or fail based on study hours

Table 2: Comparison between Linear and Logistic Regression

The fundamental distinction is that linear regression predicts a quantity (how much?), while logistic regression predicts a category (which one?). Despite sharing some mathematical foundations, they serve very different purposes in data analysis.

9 Assumptions and Limitations

9.1 Key Assumptions

Logistic regression relies on several assumptions [1]:

1. **Binary outcome**: The dependent variable should be binary (though extensions exist for multiple classes)
2. **Independence**: Observations should be independent of each other
3. **Linearity in the logit**: The log-odds should be a linear combination of the predictors
4. **No perfect multicollinearity**: Predictors should not be perfectly correlated. There should be little to no multicollinearity between independent variables
5. **Large sample size**: Generally need at least 10-15 observations per predictor
6. **No extreme outliers**: The dataset should not contain extreme outliers as they can significantly distort the coefficient estimates and affect model performance

9.2 Common Pitfalls

Multicollinearity: When predictors are highly correlated, coefficient estimates become unstable. Check for this using Variance Inflation Factor (VIF):

$$VIF_j = \frac{1}{1 - R_j^2} \tag{28}$$

where R_j^2 is the coefficient of determination when regressing x_j on all other predictors. A VIF above 5-10 indicates problematic multicollinearity.

Class Imbalance: When one class is much more frequent than the other (e.g., fraud detection where fraud is rare), the model may simply predict the majority class [10]. Solutions include:

- Resampling techniques (oversampling minority class, undersampling majority class)
- Adjusting the decision threshold
- Using different evaluation metrics (precision, recall, F1) instead of accuracy
- Applying class weights in the loss function

Overfitting: With too many predictors relative to sample size, the model may fit noise rather than signal. Use regularization or feature selection to combat this.

10 Extensions and Variants

10.1 Multinomial Logistic Regression

When the outcome has more than two categories (e.g., predicting whether a customer will buy product A, B, or C), we use multinomial logistic regression [4].

Instead of modeling a single probability, we model the probability for each of the K classes using the softmax function:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{29}$$

where K represents the number of classes and i, j iterate over all elements.

The probability that an observation belongs to class c is:

$$P(Y = c|\mathbf{X} = \mathbf{x}) = \frac{e^{\mathbf{w}_c^T \mathbf{x} + b_c}}{\sum_{k=1}^{K} e^{\mathbf{w}_k^T \mathbf{x} + b_k}} \tag{30}$$

Each class has its own set of weights $\mathbf{w}_c$ and bias b_c. The softmax ensures that all class probabilities sum to 1, making it a proper probability distribution. This formulation is widely used not only in logistic regression but also in neural networks for multi-class classification.

10.2 Regularized Logistic Regression

To prevent overfitting, we can add penalty terms to the log-likelihood [11]. Two popular approaches are:

L2 Regularization (Ridge):

$$\ell_{ridge}(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}) - \lambda \sum_{j=1}^{p} \beta_j^2 \tag{31}$$

L1 Regularization (Lasso):

$$\ell_{lasso}(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}) - \lambda \sum_{j=1}^{p} |\beta_j| \tag{32}$$

The parameter λ controls the strength of regularization. Lasso has the nice property of setting some coefficients exactly to zero, effectively performing feature selection.

Elastic Net combines both penalties:

$$\ell_{elastic}(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}) - \lambda_1 \sum_{j=1}^{p} |\beta_j| - \lambda_2 \sum_{j=1}^{p} \beta_j^2 \tag{33}$$

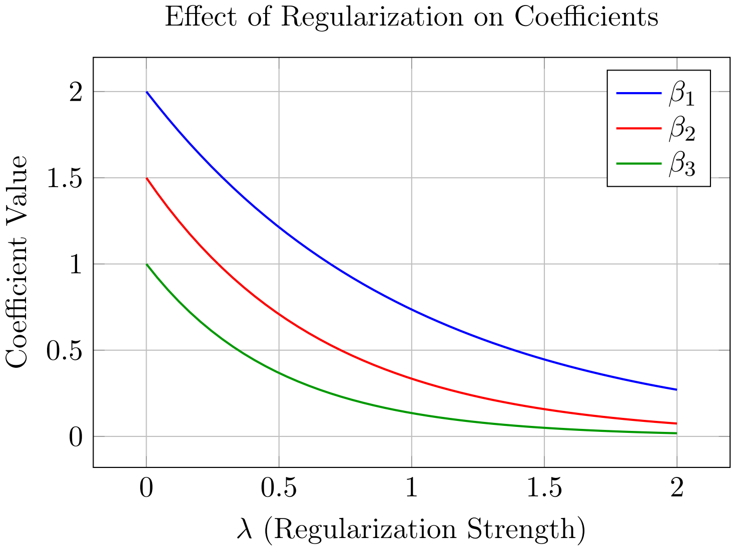

Figure 3: As regularization increases, coefficients shrink toward zero

11 Practical Applications

11.1 Healthcare: Disease Prediction

Logistic regression is widely used in medical research to predict disease risk. For example, predicting the probability of heart disease based on:

- Age
- Blood pressure
- Cholesterol level
- Smoking status
- Family history

Doctors can use these predictions to identify high-risk patients and intervene early.

11.2 Finance: Credit Scoring

Banks use logistic regression to assess credit risk. The model predicts whether a loan applicant will default based on:

- Income
- Credit history
- Debt-to-income ratio
- Employment status
- Loan amount

The interpretability of logistic regression is particularly valuable here, as lenders must explain why applications are denied.

11.3 Marketing: Customer Churn

Companies predict whether customers will cancel their subscriptions using factors like:

- Usage frequency
- Customer service interactions
- Contract length
- Payment history
- Competitor offerings

This allows proactive retention efforts targeted at at-risk customers.

11.4 Digital Marketing: Click Prediction

Online advertising platforms predict click-through rates for ads using:

- User demographics
- Browsing history
- Time of day
- Device type
- Ad characteristics

12 Conclusion

Logistic regression remains a cornerstone of statistical modeling and machine learning despite being developed decades ago. Its combination of mathematical elegance, interpretability, and practical effectiveness makes it an essential tool for anyone working with classification problems.

We've covered the theoretical foundations, from the logistic function to maximum likelihood estimation, and explored practical considerations like model evaluation, handling class imbalance, and regularization. We've also seen how logistic regression extends to multiple classes and how regularization helps prevent overfitting.

The key strengths of logistic regression are:

- Clear probabilistic interpretation
- Interpretable coefficients through odds ratios
- Computational efficiency
- Solid theoretical foundation
- Wide applicability across domains

While newer methods like random forests, gradient boosting, and neural networks can sometimes achieve better predictive performance, logistic regression often provides the best balance of accuracy, interpretability, and simplicity. It should always be considered as a baseline method before moving to more complex approaches.

As you apply logistic regression in your own work, remember that understanding your data and problem context is just as important as mastering the mathematics. The most sophisticated model is useless if it doesn't address the right question or if its predictions can't be trusted and acted upon.

Whether you're predicting disease outcomes, customer behavior, or financial risk, logistic regression provides a principled and interpretable framework for turning data into actionable insights.

References

[1] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd. Hoboken, NJ: Wiley, 2013.

[2] D. R. Cox, "The regression analysis of binary sequences," *Journal of the Royal Statistical Society: Series B*, vol. 20, no. 2, pp. 215–232, 1958.

[3] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*, 2nd. New York: Springer, 2021.

[4] A. Agresti, *Statistical Methods for the Social Sciences*, 5th. Pearson, 2018.

[5] S. Menard, *Applied Logistic Regression Analysis*, 2nd. Thousand Oaks, CA: Sage Publications, 2002.

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd. New York: Springer, 2009.

[7] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[8] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA: MIT Press, 2012.

[9] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[10] G. King and L. Zeng, "Logistic regression in rare events data," *Political Analysis*, vol. 9, no. 2, pp. 137–163, 2001.

[11] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, 2010.